Web-Based Documentation System for Dynamic Roadmaps

SAJID HASSAN AND AHSAN CHISHTI
International Islamic University Malaysia
Kuala Lampur, Malaysia
sajid.hassan@gmail.com
ahsanchisti@yahoo.com

CHANDRAN ELAMVAZUTHI
Mimos Berhad
Malaysia
chandran@mimos.my

Technology Roadmapping is a consensus-driven process to identify, evaluate, and select technology alternatives to satisfy the needs of a particular group of people. It serves as a high-level planning tool to support the development, implementation, and communication of technology development strategies and plans. The outcome of a technology roadmapping exercise is a technology roadmap, a document. Since the technology roadmap needs to be updated periodically, it remains dynamic. A documentation system that permits selective updating will be ideal. We have developed a web-based software to store, retrieve, edit, and document different types of Technology Roadmaps. In this article, we discuss the rationale for the system and describe the steps taken to develop a working system based on open source software.

Technology is one of the key sources of innovation and consequently, of value and wealth creation. Be it at the firm, industry, or national level, technology development must be planned to be able to deliver quality
products and services on time to gain a competitive advantage. This is becoming more critical as the cost, complexity, and rate of technology change increases, and the competition and sources of technology globalize. The management of technology requires effective processes and systems to ensure that the technological resources within a firm, industry or for that matter within a country are aligned with its needs, now and in the future. Roadmapping is an approach that is being increasingly applied at the firm, industry and national level to support the development, communication, and implementation of technology, and related business and policy strategies.

There are many types of roadmaps and different ways of developing them. Examples of common roadmaps include industry, technology, and product roadmaps. Industry roadmaps focus on broad market goals for a particular sector, defining a clear set of industry actions to address market, regulatory, and other barriers to growth. On the other hand, technology roadmaps identify, evaluate, and promote the development of collaborative projects within and between industries to fill technology gaps and/or capture technology related opportunities. Narrowing the scope, product roadmaps provide firms with thorough, long-range technology assessments of their future product needs and supply complete descriptions of their product lines (Schaller, pending).

In this article, we highlight the need for having a software system for storing, retrieving, editing, and documenting roadmaps. In addition we describe and discuss the steps we have taken to develop such a system.

**TECHNOLOGY ROADMAPPING**

Technology roadmapping is a consensus-based procedure to identify, evaluate and choose the technology alternatives for a selected set of needs, either at the firm, industry or national level. It is a planning process that can assist stakeholders to identify their future technology needs and the means to achieve them in a systematic manner. Said differently, it is tool to support the development, implementation and communication of strategies and plans for technology development (Phaal, Farrukh, & Probert, 2001).
Technology roadmapping can ensure that a firm or an industry can tap those critical technologies necessary to exploit opportunities from the major market developments expected to occur over a period of time. By providing strategies indicating the appropriate times to access and develop those technologies, a technology roadmap can help firms and industries to position themselves better for the future. Note that the roadmap per se is only a high-level strategy for developing identified technologies. A more detailed plan specifying the actual projects and activities is needed to operationalize the roadmap (Schaller, pending).

Operationalizing or implementing a roadmap requires resources. Hence, a gap analysis with respect to availability of resources is necessary to address implementation issues. Additional information such as current status of Research and Development (R&D) activities at the firm or industry level relevant to the technologies concerned, competence, and knowledge gaps, infrastructure, funding, and so forth, must be considered. This is important so that scarce resources can be directed more effectively. Often, projects and associated resources for the development of the technologies concerned are also identified. Hence, usually, a technology roadmap document will not only present technology strategies but also include information on their implementation (Albright, 1998; Galvin, 1998).

The communication and knowledge sharing amongst participants throughout the whole process of technology roadmapping not only leads to more effective planning but also helps to build trust and lays the foundation for possible future collaboration. Depending on how the implementation of the roadmap is orchestrated, this early build up of relationships amongst the developers of the roadmap can help to seed the development of technology clusters, which are the bedrock for sustainable innovation. Thus, technology roadmapping brings together a group of potential users to develop a framework for organizing and presenting the information to make the right technology investment decisions, and at the same time granting a way to leverage those investments (Kostoff, 1997).

There is no fixed format for describing or presenting roadmaps. Roadmaps are usually described using diagrams, charts, tables, and text or a combination of these. However, there are a few basic elements that normally should be addressed when developing a roadmap. As depicted in Figure 1, these include the future needs or problems (business, market, or societal) that the roadmap is to address; the applications (products or services) to meet those
needs or to solve those problems; the various technologies (current and future) that could possibly be used to develop the applications identified; the R&D programs and projects to develop the technologies identified; and, the resources (human, financial, etc.) required to carry out the R&D programs and projects. All these are placed in perspective with respect to a timeframe of delivery, which could be from the present to 10 years or more in future. Normally, timeframes are given in epochs, usually near-term, mid-term, and long term.

![Diagram](image)

**Figure 1.** Generic format for a roadmap (Source: T-Plan–The fast start to Technology Roadmapping, Cambridge Center for Technology Management)

**NEED FOR SOFTWARE TOOL**

During the process of developing a technology roadmap, a lot of data and information is collected for analysis and interpretation. Also, new interdependent data and information including roadmap timelines and project schedules are generated during this process. This vast amount of data and information must be stored in some form, without loss of data integrity, for easy and efficient access (Schaller, pending).
Roadmaps can be represented in different ways for example, text, tables, charts, or a combination of these. It is possible to create these representations using some of the popular word processors or presentation software. However, this can lead to discrepancy in storage and retrieval of the roadmaps. Further, this way of creating roadmap representations takes a lot more time due to a lack of appropriate editing features in these tools. Also, roadmap representations created this way are static in nature.

Technology Roadmaps by nature are dynamic; they need to be updated regularly. However, only selected portions may need to be changed. Thus, one may have access to the most recent version. Therefore, there is a need for software to help document technology roadmaps. The software should have the standard functions such as compose, edit, and delete roadmaps. The documented roadmaps should be able to be stored at a suitable location for remote access. Features such as user and group level access would also be useful. In view of all these, we have developed a software tool to aid in the documentation of technology roadmaps in a systematic and coherent manner. We have called this software *Dynamic Roadmap Documentation* system.

### DESIGN OF THE DYNAMIC ROADMAP DOCUMENTATION SOFTWARE

The Dynamic Roadmap Documentation system is also capable of converting roadmaps from a generic type to a more customized version. As mentioned earlier, the roadmaps take various forms; thus, it will be obligatory for a system to reproduce the most updated version of the roadmap in its original format. As shown in Figure 2, the roadmaps are “fed” in the system where it is constantly updated and stored and with the click of a button it can be reproduced.

The Dynamic Roadmap Documentation system resembles that of the client-server architecture. The roadmaps are stored at a central database server as depicted in Figure 3.
Figure 2. Dynamic roadmap documentation operational flow

Figure 3. Dynamic roadmap documentation system architecture
The database and the Composer reside at the server to which the client computer sends an HTTP request and is returned with an HTML query result. In other words, the client side of the system requires only a java-enabled web browser. The server-side web script (the Roadmap Documentation System), written in php language, is executed to retrieve the data from the database. The database consists of 24 interdependent tables which is the only section inaccessible in order to maintain privacy as well as security. However, the system can retrieve as well as post data into the database.

The roadmap documentation system was engineered using the open source based Hypertext Preprocessor (PHP). Since PHP supports object-oriented programming, the Documentation System is essentially an object-based system. The backbone of the Roadmap Documentation System consists of a user interface class and a project base class. All other modules and other functionalities interact as umbrella functions or derived classes from the previously mentioned base classes. The scripts are however arranged in several folders to provide a clean design. The software also makes use of several libraries, which are available under GNU public license. The script classification is shown in Figure 4.

![Figure 4. Script distribution of dynamic roadmap documentation system](image-url)
The class section contains the base class definition of the user interface as well as the individual project. All instances are created either from these classes or the derived classes in subsequent sections. Functions section mainly deals with various roles pertaining to the creation of individual roadmaps as well as assignment of jobs and tasks. This section also instantiates objects for individual roadmaps. Each object created gets a copy of all the data members from the class from which it is instantiated. Thus, we can be assured that every single object is classified suitably and the data loss in function interdependency is minimized. The image section, as the name suggests, contains the images stored in png and jpeg formats to display the figures such as background image, module images, element information image, and so forth. However, there are other aspects, which need more attention in the architecture.

The Dynamic Roadmap Documentation system uses a number of libraries (by jp group, the open source graphics group based in Norway), which provide ample support for the creation of roadmaps. These libraries provide the much needed graph drawing feature for each roadmap. The Style and Preferences sections as the names suggest, provides the preferred values for each individual user in terms of choice of color, number, and date format. In addition, this section also stores the security data for each user.

The architecture of the system is shown in Figure 5. Whenever the user provides the login name and the appropriate password, the system will create a set of objects from the base classes. The instance, which will be created from the user interface class will then retrieve the data corresponding to this particular user from the database. The retrieved data include security information, that is, the login name, password, and so forth. The data also includes the type of user for example, administrator, normal user, moderator, and so forth. In addition, the preferences for the particular user are also retrieved, that is, the color choice, the currency format, forum news, and private messages.
Once the user selects to compose a new roadmap document, the system creates objects to be used for generating roadmaps. These objects primarily are formed from the Project base class, which interact with certain libraries. If the roadmap document has been previously created, then the system also retrieves the individual element data. The last thing the objects capture is the preference value for each roadmap. The preferences include the color code, user accessibility, and the roadmap format.

A typical roadmap would have features such as individual elements, their immediate and previous steps, tasks, and the time schedule. As an example a roadmap featuring Smart Human Environment is shown in Figure 6. This roadmap features the timeline and the individual elements for this particular roadmap.

The previously-mentioned Smart Human Environment roadmap was created using presentation software. The same roadmap if documented under the Dynamic Roadmap Documentation system will appear similar to Figure 7.

**Figure 5.** Software architecture
Figure 6. Smart home environment roadmap (Source: VTT technical research centre of Finland)

Figure 7. A typical screenshot from roadmap documentation system showing smart home environment roadmap
SOFTWARE TOOLS USED

The software tools that were used for the development of Dynamic Roadmap Documentation system are:

1. Apache web server: It is an open source web server which is available for free at the internet from the following website http://www.apache.org/download. It is compatible with both UNIX and Windows environment.

2. MySQL web database server: In addition to the web server, a database server was needed. Since, the whole architecture is designed to be an open source system, the database server is also an open-source one available at http://www.mysql.org/

3. php: php is a server side web scripting language and it is also open source. The php can be configured automatically or manually to meet requirements.

4. Microsoft Frontpage 2000: Although the backbone of the system is php and javascript, to accomplish some HTML programming, Frontpage 2000 was used.

CONCLUSION AND FUTURE WORK

The Dynamic Roadmap Documentation system is a useful tool to help document technology roadmaps in a more efficient and effective manner. It can also be used to manage the projects that arise through the operationalization of a roadmap. Further, it can be used to develop communities of practice of roadmap users using features such as discussion forums and personal messages that have been provided. This could be enhanced through the provision of a real-time collaborative tool, where users can have real-time conversations.
References


